This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently Amended) A system for providing data from a graphical code reading device to a software module, the system comprising:

a graphical code reading device that scans a graphical code and provides graphical code data; and

a computer in electronic communication with the graphical code reading device that receives the graphical code data from the graphical code reading device, the computer comprising:

a processor;

a memory in electronic communication with the processor;

a communications port in electronic communication with the processor for communicating with the graphical code reading device;

an association including a data type and a software module identifier, wherein the data type indicates what type of data is included in the graphical code data; and

a driver that receives the graphical code data and obtains the software module identifier through use of the data type and provides the graphical code data to the software module identified by the software module identifier.

2. (Original) The system as defined in claim 1 wherein the association is included in an association list.

3. (Original) The system as defined in claim 2 wherein the association list is created statically.

4. (Original) The system as defined in claim 2 wherein the association list is created dynamically.

5. (Original) The system as defined in claim 1 wherein the driver includes a server module.

6. (Original) The system as defined in claim 5 wherein the server module implements a method comprising:

receiving the graphical code data;

determining a data type of the graphical code data;

obtaining the software module identifier from the association through use of the data type;

launching the software module identified by the software module identifier if the software module is not running; and

providing the graphical code data to a client module of the software module.

7. (Original) The system as defined in claim 6 wherein the data type of the graphical code data is determined according to the symbology of the graphical code.

8. (Original) The system as defined in claim 6 wherein the data type of the graphical code data is determined by examining graphical code data content.

9. (Original) The system as defined in claim 1 wherein the graphical code data includes a plurality of data fields.

10. (Original) The system as defined in claim 9 wherein one or more of the data fields includes a data type identifier.

11. (Original) The system as defined in claim 1 wherein the graphical code data includes a data type identifier.

12. (Original) The system as defined in claim 1 wherein the association is included in an association list comprising a plurality of data types and a plurality of software module identifiers.

13. (Currently Amended) A computer system for providing data from a graphical code reading device to a software module, the computer system comprising:

a processor;

a memory in electronic communication with the processor;

a communications port in electronic communication with the processor for communicating with a graphical code reading device;

an association including a data type and a software module identifier, wherein the data type indicates what type of data is included in the graphical code data; and

a driver that receives graphical code data from the graphical code reading device through the communications port and obtains the software module identifier through use of the data type and provides the graphical code data to the software module identified by the software module identifier.

14. (Original) The computer system as defined in claim 13 wherein the association is included in an association list.

15. (Original) The computer system as defined in claim 14 wherein the association list is created statically.

16. (Original) The computer system as defined in claim 14 wherein the association list is created dynamically.

17. (Original) The computer system as defined in claim 13 wherein the driver includes a server module.

18. (Original) The computer system as defined in claim 17 wherein the server module implements a method comprising:

receiving the graphical code data;

determining a data type of the graphical code data;

obtaining the software module identifier from the association through use of the data type;

launching the software module identified by the software module identifier if the software module is not running; and

providing the graphical code data to a client module of the software module.

19. (Original) The computer system as defined in claim 13 wherein the graphical code data includes a plurality of data fields.

20. (Original) The computer system as defined in claim 18 wherein the data type of the graphical code data is determined according to the symbology of the graphical code.

21. (Original) The computer system as defined in claim 18 wherein the data type of the graphical code data is determined by examining graphical code data content.

22. (Original) The computer system as defined in claim 19 wherein one or more of the data fields includes a data type identifier.

23. (Original) The computer system as defined in claim 13 wherein the graphical code data includes a data type identifier.

24. (Original) The computer system as defined in claim 13 wherein the association is included in an association list comprising a plurality of data types and a plurality of software module identifiers.

25. (Currently Amended) A computer-readable medium for storing program data, wherein the program data comprises executable instructions for implementing a method comprising:

receiving graphical code data from a graphical code reading device in electronic communication with a computer;

determining a data type of the graphical code data, wherein the data type indicates what type of data is included in the graphical code data;

obtaining a software module identifier through use of the data type and through use of an association list;

launching a software module identified by the software module identifier if the software module is not running; and

providing the graphical code data to the software module.

26. (Original) The medium as defined in claim 25 wherein the association list is created statically.

27. (Original) The medium as defined in claim 25 wherein the association list is created dynamically.

28. (Original) The medium as defined in claim 25 wherein the graphical code data includes a plurality of data fields.

29. (Original) The medium as defined in claim 25 wherein the data type of the graphical code data is determined according to the symbology of the graphical code.

30. (Original) The medium as defined in claim 25 wherein the data type of the graphical code data is determined by examining graphical code data content.

31. (Original) The medium as defined in claim 28 wherein one or more of the data fields includes a data type identifier.

32. (Original) The medium as defined in claim 25 wherein the graphical code data includes a data type identifier.

33. (Original) The medium as defined in claim 25 wherein the association list comprises a plurality of data types and a plurality of software module identifiers.

34. (Currently Amended) A method for providing data from a graphical code reading device to a software module, the method comprising:

receiving graphical code data from a graphical code reading device in electronic communication with a computer;

determining a data type of the graphical code data, wherein the data type indicates what type of data is included in the graphical code data;

obtaining a software module identifier through use of the data type and through use of an association list;

launching a software module identified by the software module identifier if the software module is not running; and

providing the graphical code data to the software module.

35. (Original) The method as defined in claim 34 wherein the association list is created statically.

36. (Original) The method as defined in claim 34 wherein the association list is created dynamically.

37. (Original) The method as defined in claim 34 wherein the graphical code data includes a plurality of data fields.

38. (Original) The method as defined in claim 34 wherein the data type of the graphical code data is determined according to the symbology of the graphical code.

39. (Original) The method as defined in claim 34 wherein the data type of the graphical code data is determined by examining graphical code data content.

40. (Original) The method as defined in claim 37 wherein one or more of the data fields includes a data type identifier.

41. (Original) The method as defined in claim 34 wherein the graphical code data includes a data type identifier.

42. (Original) The method as defined in claim 34 wherein the association list comprises a plurality of data types and a plurality of software module identifiers.

43. (Currently Amended) A driver for communicating with a graphical code reading device in electronic communication with a computer, the driver comprising:

a server module implementing a method comprising:

receiving graphical code data from the graphical code reading device;

determining a data type of the graphical code data, wherein the data type indicates what type of data is included in the graphical code data;

accessing an association list comprising a data type list and a software module list;

obtaining a software module identifier from the association list through use of the data type and the data type list;

launching the software module identified by the software module identifier if the software module is not running; and

providing the graphical code data to a client module of the software module.

44. (Original) The driver as defined in claim 43 wherein the association list is created statically.

45. (Original) The driver as defined in claim 43 wherein the association list is created dynamically.

46. (Original) The driver as defined in claim 43 wherein the graphical code data includes a plurality of data fields.

47. (Original) The driver as defined in claim 43 wherein the data type of the graphical code data is determined according to the symbology of the graphical code.

48. (Original) The driver as defined in claim 43 wherein the data type of the graphical code data is determined by examining graphical code data content.

49. (Original) The driver as defined in claim 43 wherein one or more of the data fields includes a data type identifier.

50. (Original) The driver as defined in claim 43 wherein the graphical code data includes a data type identifier.

51. (Currently Amended) A system for providing data from a graphical code reading device to a software module, the system comprising:

a graphical code reading device that scans a graphical code and provides graphical code data; and

a computer in electronic communication with the graphical code reading device that receives the graphical code data from the graphical code reading device, the computer comprising:

a processor;

a memory in electronic communication with the processor;

a communications port in electronic communication with the processor for communicating with the graphical code reading device;

an association list including a plurality of data types and a plurality of software module identifiers;

a first software module including a client module; and

a driver that includes a server module, the server module implementing a method comprising:

receiving the graphical code data;

determining a data type of the graphical code data, wherein the data type indicates what type of data is included in the graphical code data;

obtaining a first software module identifier from the association list through use of the data type;

launching the first software module identified by the first software module identifier if the first software module is not running; and

providing the graphical code data to the client module of the first software module.

52. (Currently Amended) A system for providing data from a graphical code reading device to a web page, the system comprising:

a graphical code reading device that scans a graphical code and provides graphical code data; and

a computer in electronic communication with the graphical code reading device that receives the graphical code data from the graphical code reading device, the computer comprising:

a processor;

a memory in electronic communication with the processor;

a communications port in electronic communication with the processor for communicating with the graphical code reading device;

an association including a data type and a software module identifier, wherein the data type indicates what type of data is included in the graphical code data; and

a driver that receives the graphical code data and obtains the software module identifier through use of the data type and provides the graphical code data to a browser extension module identified by the software module identifier and wherein the browser extension module provides the graphical code data to the web page.

53. (Original) The system as defined in claim 52 wherein the web page includes a form having form fields.

54. (Original) The system as defined in claim 53 wherein the graphical code data comprises a plurality of data fields that correspond to the form fields and whereby the form is automatically completed.

55. (Original) The system as defined in claim 52 wherein the driver providing the graphical code data to the browser extension module is accomplished by the driver sending the graphical

code data to a client module embedded in the browser extension module and by the client module calling a browser extension callback function to pass the graphical code data to the browser extension module.

56. (Original) The system as defined in claim 52 wherein the browser extension module provides the graphical code data to the web page by calling a callback function in a script contained in the web page.

57. (Original) The system as defined in claim 52 wherein the browser extension module provides the graphical code data to the web page by posting an event to the web page, wherein the web page contains a script that responds to the event, and wherein the script obtains the graphical code data by calling a function within the browser extension module.

58. (Original) The system as defined in claim 52 wherein the browser extension module provides the graphical code data to the web page by posting an event to the web page, the event including the graphical code data, wherein the web page contains a script that responds to the event.

59. (Original) The system as defined in claim 52 wherein a serialization process is used to avoid losing transactions.

60. (Original) The system as defined in claim 59 wherein the serialization process includes causing the driver to wait until a message is sent from the client module to stop the wait.

61. (Original) The system as defined in claim 52 wherein the browser extension module is started by a tag embedded in the web page.

62. (Currently Amended) A computer system for providing data from a graphical code reading device to a web page, the computer system comprising:

a processor;

a memory in electronic communication with the processor;

a communications port in electronic communication with the processor for communicating with a graphical code reading device;

an association including a data type and a software module identifier, wherein the data type indicates what type of data is included in the graphical code data; and

a driver that receives graphical code data from the graphical code reading device through the communications port and obtains the software module identifier through use of the data type and provides the graphical code data to a browser extension module identified by the software module identifier, the browser extension module operating to provide the graphical code data to the web page.

63. (Original) The computer system as defined in claim 62 wherein the web page includes a form having form fields.

64. (Original) The computer system as defined in claim 63 wherein the graphical code data comprises a plurality of data fields that correspond to the form fields and whereby the form is automatically completed.

65. (Original) The computer system as defined in claim 62 wherein the driver providing the graphical code data to the browser extension module is accomplished by the driver sending the graphical code data to a client module embedded in the browser extension module and by the client module calling a browser extension callback function to pass the graphical code data to the browser extension module.

66. (Original) The computer system as defined in claim 62 wherein the browser extension module provides the graphical code data to the web page by calling a callback function in a script contained in the web page.

67. (Original) The computer system as defined in claim 62 wherein the browser extension module provides the graphical code data to the web page by posting an event to the web page, wherein the web page contains a script that responds to the event, and wherein the script obtains the graphical code data by calling a function within the browser extension module.

68. (Original) The computer system as defined in claim 62 wherein the browser extension module provides the graphical code data to the web page by posting an event to the web page, the event including the graphical code data, wherein the web page contains a script that responds to the event.

69. (Original) The computer system as defined in claim 62 wherein a serialization process is used to avoid losing transactions.

70. (Original) The computer system as defined in claim 69 wherein the serialization process includes causing the driver to wait until a message is sent from the client module to stop the wait.

71. (Original) The computer system as defined in claim 62 wherein the browser extension module is started by a tag embedded in a web page.

72. (Currently Amended) A method for providing data from a graphical code reading device to a web page, the method comprising:

receiving graphical code data from a graphical code reading device in electronic communication with a computer;

determining a data type of the graphical code data, wherein the data type indicates what type of data is included in the graphical code data;

obtaining a software module identifier through use of the data type and through use of an association list;

launching a web browser identified by the software module identifier if the web browser is not running; and

providing the graphical code data to a web page opened by the web browser.

73. (Original) The method as defined in claim 72 wherein the web page includes a form having form fields.

74. (Original) The method as defined in claim 73 further comprising completing the form through use of the graphical code data that further comprises a plurality of data fields that correspond to the form fields.

75. (Original) The method as defined in claim 72 further comprising sending the graphical code data to a client module embedded in a browser extension module.

76. (Original) The method as defined in claim 75 further comprising passing the graphical code data to the browser extension module by calling a browser extension callback function.

77. (Original) The method as defined in claim 75 wherein the browser extension module provides the graphical code data to the web page by calling a callback function in a script contained in the web page.

78. (Original) The method as defined in claim 75 wherein the browser extension module provides the graphical code data to the web page by posting an event to the web page, wherein

the web page contains a script that responds to the event, and wherein the script obtains the graphical code data by calling a function within the browser extension module.

79. (Original) The method as defined in claim 75 wherein the browser extension module provides the graphical code data to the web page by posting an event to the web page, the event including the graphical code data, wherein the web page contains a script that responds to the event.

80. (Currently Amended) A system for providing data from a graphical code reading device to an extensible application, the system comprising:

a graphical code reading device that scans a graphical code and provides graphical code data; and

a computer in electronic communication with the graphical code reading device that receives the graphical code data from the graphical code reading device, the computer comprising:

a processor;

a memory in electronic communication with the processor;

a communications port in electronic communication with the processor for communicating with the graphical code reading device;

an association including a data type and a software module identifier, wherein the data type indicates what type of data is included in the graphical code data; and

a driver that receives the graphical code data and obtains the software module identifier through use of the data type and provides the graphical code data to an extension module of the extensible application identified by the software module identifier.

81. (Original) The system as defined in claim 80 wherein the extensible application is a web browser and wherein the extension module is a browser extension module.

82. (Original) The system as defined in claim 80 wherein the graphical code data comprises a plurality of data fields.

83. (Original) The system as defined in claim 80 wherein the driver providing the graphical code data to the extension module is accomplished by the driver sending the graphical code data to a client module embedded in the extension module.

84. (Original) The system as defined in claim 83 wherein a serialization process is used to avoid losing transactions.

85. (Original) The system as defined in claim 84 wherein the serialization process includes causing the driver to wait until a message is sent from the client module to stop the wait.

86. (Currently Amended) A computer system for providing data from a graphical code reading device to an extensible application, the computer system comprising:

a processor;

a memory in electronic communication with the processor;

a communications port in electronic communication with the processor for communicating with a graphical code reading device;

an association including a data type and a software module identifier, wherein the data type indicates what type of data is included in the graphical code data; and

a driver that receives graphical code data from the graphical code reading device through the communications port and obtains the software module identifier through use of the data type and provides the graphical code data to an extension module identified by the software module identifier.

87. (Original) The computer system as defined in claim 86 wherein the extensible application is a web browser and wherein the extension module is a browser extension module.

88. (Original) The computer system as defined in claim 86 wherein the graphical code data comprises a plurality of data fields.

89. (Original) The computer system as defined in claim 86 wherein the driver providing the graphical code data to the extension module is accomplished by the driver sending the graphical code data to a client module embedded in the extension module.

90. (Original) The computer system as defined in claim 89 wherein a serialization process is used to avoid losing transactions.

91. (Original) The computer system as defined in claim 90 wherein the serialization process includes causing the driver to wait until a message is sent from the client module to stop the wait.

92. (Currently Amended) A method for providing data from a graphical code reading device to an extensible application, the method comprising:

receiving graphical code data from a graphical code reading device in electronic communication with a computer;

determining a data type of the graphical code data, wherein the data type indicates what type of data is included in the graphical code data;

obtaining a software module identifier through use of the data type and through use of an association list;

launching the extensible application identified by the software module identifier if the extensible application is not running; and

providing the graphical code data to an extension module of the extensible application.

93. (Original) The method as defined in claim 92 wherein the graphical code data further comprises a plurality of data fields.

94. (Original) The method as defined in claim 92 wherein the data type of the graphical code data is determined according to the symbology of the graphical code.

95. (Original) The method as defined in claim 92 wherein the data type of the graphical code data is determined by examining graphical code data content.

96. (Original) The method as defined in claim 93 wherein one or more of the data fields includes a data type identifier.

97. (Original) The method as defined in claim 92 wherein the graphical code data includes a data type identifier.

98. (Original) The method as defined in claim 92 further comprising sending the graphical code data to a client module embedded in the extension module.